SYSTEM AND METHOD FOR SEARCHING AND RECOMMENDING OBJECTS FROM A CATEGORICALLY ORGANIZED INFORMATION REPOSITORY

CROSS-REFERENCE TO RELATED APPLICATIONS

5   This application is a continuation-in-part of U.S. Patent Application Serial No. 09/305,844, filed May 5, 1999, entitled SYSTEM AND METHOD FOR SEARCH AND RECOMMENDING DOCUMENTS IN A COLLECTION USING SHARED BOOKMARKS, and is related to co-pending application entitled HIERARCHICAL MODEL FOR DYNAMIC PERSONALIZATION OF OBJECT SPACES AND PRIVACY PROTECTION, filed on instant date herewith, both of

10   which are commonly owned by the assignee of the present invention and the entire disclosures of which are expressly incorporated herein by reference.

FIELD OF THE INVENTION

       The invention relates to the field of information searching and browsing, and more particularly to a system and method for enhancing searches and recommending documents in a

15   collection through the use of bookmarks shared among a community of users.

BACKGROUND OF THE INVENTION

       Computer users are increasingly finding navigating document collections to be difficult because of the increasing size of such collections. For example, the World Wide Web on the Internet includes millions of individual pages. Moreover, large companies' internal Intranets often include

20   repositories filled with many thousands of documents.

1

It is frequently true that the documents on the Web and in Intranet repositories are not very well indexed. Consequently, finding desired information in such a large collection, unless the identity, location, or characteristics of a specific document are well known, can be much like looking for a needle in a haystack.

5   The World Wide Web is a loosely interlinked collection of documents (mostly text and images) located on servers distributed over the Internet. Generally speaking, each document has an address, or Uniform Resource Locator (URL), in the exemplary form "http://www.server.net/directory/file.html". In that notation, the "http:" specifies the protocol by which the document is to be delivered, in this case the "HyperText Transport Protocol." The 10 "www.server.net" specifies the name of a computer, or server, on which the document resides; "directory" refers to a directory or folder on the server in which the document resides; and "file.html" specifies the name of the file.

Most documents on the Web are in HTML (HyperText Markup Language) format, which allows for formatting to be applied to the document, external content (such as images and other 15 multimedia data types) to be introduced within the document, and "hotlinks" or "links" to other documents to be placed within the document, among other things. "Hotlinking" allows a user to navigate between documents on the Web simply by selecting an item of interest within a page. For example, a Web page about reprographic technology might have a hotlink to the Xerox corporate web site. By selecting the hotlink (often by clicking a marked word, image, or area with a pointing 20   device, such as a mouse), the user's Web browser is instructed to follow the hotlink (usually via a URL, frequently invisible to the user, associated with the hotlink) and read a different document.

2

Obviously, a user cannot be expected to remember a URL for each and every document on the Internet, or even those documents in a smaller collection of preferred documents. Accordingly, navigation assistance is not only helpful, but necessary.

Modern Web browsers (software applications used to view and navigate documents on the Web) have introduced the concept of "bookmarks" or "favorites" (collectively referred to as "bookmarks" in this document). Bookmarks allow a user to identify which documents he would like to keep track of. The user's local machine then keeps track of the URLs for those sites, allowing the user to reload and view the sites' contents at any desired time. Bookmarks can be thought of as "pointers" to content on the Web, each specifying an address that identifies the location of the desired document, but not including the document's content (except, perhaps, in a descriptive title of the document).

In current versions of Netscape Navigator (specifically, at least versions 3.x and 4.x), a user's bookmarks are stored and maintained in a special HTML file stored on the user's local machine. This file includes a list of sites represented as title and URL pairs (in a user-defined hierarchy, if desired). The user's entire set of bookmarks is contained within a single HTML file.

Recent versions of Microsoft's Internet Explorer (at least versions 3.x-5.x) store user bookmarks (or "favorites," using Microsoft's preferred terminology) as individual files on the local machine's file system. Each favorite is a small file containing the site's URL, while the favorite's title is stored as the filename.

Other browsers' bookmarks are frequently stored as entries in a custom configuration file, in which each site's title is paired with a URL.

3

None of the foregoing browsers permit much sophisticated user of a user's collection of bookmarks, although some limited manipulations are possible. For example, it is usually possible to create and modify a hierarchy of bookmarks (including sorting and moving existing bookmarks around within the hierarchy); to modify the titles paired with the URLs, to search for words within

5   the titles or URLs; and often to derive some additional information about the bookmarks, such as the date and time of the user's most-recent visit to the site, the collected number of visits, and possibly other information.

In typical use, the bookmark facilities of Web browsers act as a "filter" for those documents a particular user finds to be important or useful. While a user might view hundreds of Web pages in a

10 day, only a few of those are typically found to provide useful information. If that information is expected to be useful again in the future, the user will often set a bookmark for those pages. This is a useful way for users to be able to access the Internet; however, traditional bookmarks have the distinct limitation that they are only useful to the extent a user has seen the sites before, since adding a bookmark to a collection is a manual act, typically performed when either the desired page is

15 already being viewed or a URL has been manually received from another person.

Most notably, the known traditional bookmark systems are single-user. Of course (particularly with Netscape Navigator, in which bookmarks already exist in an HTML file), bookmarks can be exported to a public web page, allowing others to view and use the bookmarks, but that in itself does not provide any additional functionality.

20 Accordingly, when a user desires to find information on the Internet (or other large network) that is not already represented in the user's bookmark collection, the user will frequently turn to a

"search engine" to locate the information. A search engine serves as an index into the content stored on the Internet.

There are two primary categories of search engines: those that include documents and Web sites that are analyzed and used to populate a hierarchy of subject-matter categories (e.g., Yahoo),

5   and those that "crawl" the Web or document collections to build a searchable database of terms, allowing keyword searches on page content (such as AltaVista, Excite, and Infoseek, among many others).

Also known are recommendation systems, which are capable of providing Web site recommendations based on criteria provided by a user or by comparison to a single preferred

10   document (e.g., Firefly, Excite's "more like this" feature).

"Google" (www.google.com) is an example of a search engine that incorporates several recommendation-system-like features. It operates in a similar manner to traditional keyword-based search engines, in that a search begins by the user's entry of one or more search terms used in a pattern-matching analysis of documents on the Web. It differs from traditional keyword-based

15   search engines (such as AltaVista), in that search results are ranked based on a metric of page "importance," which differs from the number of occurrences of the desired search terms (and simple variations upon that theme).

Google's metric of importance is based upon two primary factors: the number of pages (elsewhere on the Web) that link to a page (i.e., "inlinks," defining the retrieved page as an

20   "authority"), and the number of pages that the retrieved page links to (i.e., "outlinks," defining the retrieved page as a "hub"). A page's inlinks and outlinks are weighted, based on the Google-determined importance of the linked pages, resulting in an importance score for each retrieved page.

The search results are presented in order of decreasing score, with the most important pages presented first. It should be noted that Google's page importance metric is based on the pattern of links on the Web as a whole, and is not limited (and at this time cannot be limited) to the preferences of a single user or group of users.

5   Another recent non-traditional search engine is IBM's CLEVER (CLient-side EigenVector Enhanced Retrieval) system. CLEVER, like Google, operates like a traditional search engine, and uses inlinks/authorities and outlinks/hubs as metrics of page importance. Again, importance (based on links throughout the Web) is used to rank search results. Unlike Google, CLEVER uses page content (e.g., the words surrounding inlinks and outlinks) to attempt to classify a page's subject

10   matter. Also, CLEVER does not use its own database of Web content; rather, it uses an external hub, such as an index built by another search engine, to define initial communities of documents on the Web. From hubs on the Web that frequently represent people's interests, CLEVER is able to identify communities, and from those communities, identify related or important pages.

        Direct Hit is a service that cooperates with traditional search engines (such as HotBot),

15   attempting to determine which pages returned in a batch of results are interesting or important, as perceived by users who have previously performed similar searches. Direct Hit tracks which pages in a list of search results are accessed most frequently; it is also able to track the amount of time users spend at the linked sites before returning to the search results. The most popular sites are promoted (i.e., given higher scores) for future searches.

20   Alexa is a system that is capable of tracking a user's actions while browsing. By doing so, Alexa maintains a database of users' browsing histories. Page importance is derived from other users' browsing histories. Accordingly, at any point (not just in the context of a search), Alexa can

6

provide a user with information on related pages, derived from overall traffic patterns, link structures, page content, and editorial suggestions.

Knowledge Pump, a Xerox system, provides community-based recommendations by initially allowing users to identify their interests and "experts" in the areas of those interests. Knowledge

5   Pump is then able to "push" relevant information to the users based on those preferences; this is accomplished by monitoring network traffic to create profiles of users, including their interests and "communities of practice," thereby refining the community specifications. However, Knowledge Pump does not presently perform any enhanced search and retrieval actions like the search-engine-based systems described above.

10   While the foregoing systems and services blend traditional search engine and recommendation system capabilities to some degree, it should be recognized that none of them are presently adaptable to provide search-engine-like capabilities while taking into account the preferences of a smaller group than the Internet as a whole. In particular, it would be beneficial to be able to incorporate community-based recommendations into a system that is capable of retrieving

15   previously unknown documents from the Internet.

SUMMARY OF THE INVENTION

The present system and method facilitate searching and recommending resources, or documents, based upon a collection of user document preferences shared by a large group of users. The invention leverages several of the key properties of document collections: only valuable

20   documents are bookmarked; documents are usually categorized into a hierarchy; and documents can be shared. In a preferred embodiment, the present system combines some attributes of bookmark

systems, as discussed above, with some attributes of search engines and recommendation systems, also discussed above.

The present system and method maintain a centralized database of bookmarks or user document preferences. This centralized database is maintained as a hierarchy, with individual users'

5   bookmarks maintained separately from other users' bookmarks. However, the maintenance of the centralized database facilitates harnessing the power and flexibility of being able to use, in various ways, all users' public bookmarks and the information contained in and referenced by those bookmarks.

The system and method of the present invention allows for several operations to be

10   performed, including enhanced search and retrieval, enhanced subject-matter-based recommendation generation (for both documents and groups), and automatic document categorization and summarization.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other features, aspects and advantages of the present invention will be more fully

15   understood when considered in connection with the following specification, appended claims, and accompanying drawings, wherein:

FIGURE 1 is a block diagram illustrating the physical aspects of the invention, namely several users in communication with the Internet;

FIGURE 2 is a representation of the user interface presented by the main bookmark window

20   of a system according to the invention;

8

FIGURE 3 is an alternate view of the user interface of FIGURE 2, in which the command entry listbox is expanded to show several available commands for the system;

FIGURE 4 is a representation of the user interface presented by a search results window of a system according to the invention;

5 FIGURE 5 is a representation of the user interface presented by an "Add New Bookmark" window of a system according to the invention;

FIGURE 6 is a representation of the user interface presented by a bookmark editing window of a system according to the invention;

FIGURE 7 is a representative schematic diagram illustrating the relationships among all 10 possible users of a system according to the invention, groups selected from the users, and individual users;

FIGURE 8 is a representative schematic diagram illustrating the relationships among all possible documents on the Internet, a viewed selection of documents, and a typical set of bookmarked documents;

15 FIGURE 9 is a block diagram illustrating the functional components and communications of a server-based implementation of the present invention;

FIGURE 10 is a block diagram illustrating the functional components and communications of a client-based implementation of the present invention;

FIGURE 11 is a flow chart illustrating the sequence of steps performed by a user's machine 20 in the context of the server-based implementation of FIGURE 9;

FIGURE 12 is a flow chart illustrating the sequence of steps performed by a user's machine in the context of the client-based implementation of FIGURE 10;

9

FIGURE 13 is a block diagram illustrating the background processing typically performed by the bookmark database in a system according to the invention;

FIGURE 14 is a flow chart illustrating the sequence of steps performed in generating a subject-matter recommendation in a system according to the invention;

5   FIGURE 15 is a flow chart illustrating the sequence of steps performed in generating an augmented search with ranked results in a system according to the invention;

FIGURE 16 is a block diagram illustrating the functional components and communication flow of a first embodiment of the inventive system operating in conjunction with a search engine;

FIGURE 17 is a block diagram illustrating the functional components and communication

10   flow of a second embodiment of the system according to the invention operating in conjunction with a search engine;

FIGURE 18 is a flow chart illustrating the sequence of steps performed in generating an augmented search with contextually ranked results in accordance with the system of Figure 16; and

FIGURE 19 is a flow chart illustrating the sequence of steps performed in generating an

15   augmented search and contextually ranked results in accordance with the system of Figure 17.

DETAILED DESCRIPTION OF THE INVENTION

The invention is described below, with reference to detailed illustrative embodiments. It will be apparent that the invention can be embodied in a wide variety of forms, some of which may be quite different from those of the disclosed embodiments. Consequently, the specific structural and

20   functional details disclosed herein are merely representative and do not limit the scope of the invention.

A bookmark system according to the present invention can be implemented as set forth in Fig. 1. The exemplary system includes at least one user 110, and typically a plurality of users 110, 112, 114, and 116. Each user 110-116 is coupled to a document repository 118, such as the Internet (or the World Wide Web), a corporate Intranet, a library, or any other collection of documents.

5 Documents, in this context, refers to any data file containing information readable by a machine or a human; the term includes (but is not limited to) text files, formatted text files, bitmapped image files (including images representing text-based documents), vector-based image files, sound files, multimedia files, and any other data files of possible interest. Documents may be static or dynamically generated based on information in the URL or other request presented to access the

10 document as well as other contextual information, such as the time of day. The repository 118 may be situated on distributed servers such as Web servers on the Internet, a single group of dedicated servers such as a corporate information center, or a single host server.

Also in communication with the repository 118 (and the users 110-116) is a bookmark database 120. As described in the summary set forth above, the bookmark database 120 maintains a

15 set of bookmarks for each user 110-116; each user may maintain private bookmarks, which are shielded by the database 120 from the other users, as well as public bookmarks, which are made available by the database 120 for various uses by other users. The specific operation of the bookmark database 120 will be set forth in additional detail below.

A user's primary interaction with the bookmark database 120 takes place through a main

20 bookmark window 210 (Fig. 2) provided by the user's browser 122 (Fig. 1). The illustrated bookmark window 210 includes a list of bookmarks 212, a list of categories 213, the user's screen

11

name "cwixon" 214, and several cosmetic separators 216. A data entry area 218 is also provided, and will be discussed in further detail below.

The user 110 can access bookmarks within the list 212 in several possible ways. By "clicking" on the text of a single bookmark (e.g., the White House bookmark 220), the user's

5   browser 122 will open a new window and bring up the document referenced by the bookmark 220, in this case the White House's Web site. By "dragging" the bookmark 220 into an existing browser window, the browser 122 will bring up the document in that window. In either case, the system, in its preferred embodiment, logs the event of accessing the bookmark 220 to facilitate tracking frequency and recency of use for all bookmarks.

10   In the left side of the main bookmark window 210 next to the list of bookmarks 212, there are three columns 222, 224, and 226 for informational icons representing whether the documents corresponding to each bookmark are available, relatively new, or popular. In the illustrated embodiment of the invention, the first column 222 is reserved for an icon representing that a particular document is presently unavailable. For example, an unavailability icon 228 is shown next

15   to the "This is a Bad Link" bookmark; the system is able to alert the user that no such document exists without the need for the user to manually verify the site's availability. In one embodiment of the invention, the system is able to recommend alternative documents when a preferred document is unavailable; that capability will be discussed in further detail below.

The second column 224 is reserved for an icon representing that a particular document is new

20   or has been revised within the last thirty days. For example, a newness icon 230 is shown next to the Infoseek bookmark; the presence of that icon in the presently implemented and illustrated embodiment tells the user that the Infoseek main page (the one referenced by the bookmark) has been

12

updated recently, within the last thirty days. In an alternative embodiment of the invention, the newness icon 230 can be used in combination with the recency-of-access tracking performed by the invention, thereby alerting the user that a document has been updated since the user last accessed or viewed the document.

5   The third column 226 is reserved for an icon representing the popularity of a particular document. In the illustrated embodiment, a popularity icon 232 is shown next to the AltaVista bookmark because that bookmark is present in over 50% of all users' bookmark collections. The popularity icon 232 can thus be an indication of the quality of the information likely to be found in the document, since other users are also relying upon it. Although the currently implemented and illustrated embodiment uses 50% (over all users) as the popularity threshold, it should be noted that other thresholds might also be usable, as this number is used simply for convenience. In alternative embodiments of the invention, the threshold percentage may be made a user-definable preference, or may be based on some other metric (such as the percentage over a specified group of users).

In a preferred embodiment of the invention, the meaning of each icon is explained via a pop-up message when the user's mouse pointer (or other selection tool) is positioned over the icon. For example, by positioning the mouse pointer over the unavailability icon 228, a "Document Unavailable" message would be presented to the user.

The icons in columns 222, 224, and 226 are presented to each user on the basis of information tracked and maintained by the bookmark database 120. The database 120 tracks availability and newness for each bookmark in the background, without user intervention, to achieve some efficiencies that would not otherwise be possible when each user maintains a separate local collection of bookmarks. Specifically, and particularly in the case of popular bookmarks, the

13

information on availability and newness can be updated for the benefit of plural users at one time through one simple attempted access operation by the bookmark database 120. For example, it is likely that many (or even all) of the users of the system will have a bookmark for "Yahoo!" The availability and newness of that site can be tracked once by the database 120; the information can

5   then be propagated to each user through multiple instances of the main bookmark window 210.

The background operations of the database 120 will be discussed in further detail below, in connection with Fig. 13.

As suggested above, a user's bookmarks can be divided into categories (such as the categories in the list 213); the list 212 shown in Fig. 2 is the "top level" list in user cwixon' s

10   hierarchy, as illustrated by the "slash" 234 presented after the user's screen name 214. This is consistent with the traditional Unix-like method for specifying hierarchies of directories, in which a slash character alone represents the root or top-level directory. The user 110 can access the other categories in the list 213 by clicking on those category names. When that is done, the list of bookmarks 212 will be replaced with a different list taken from the chosen category (such as News or

15   Weather in the illustrated example), and the slash 234 will be followed by the name of the chosen category. After descending the hierarchy, the user 110 can return to the top level (or any intermediate level) by clicking on the screen name 214 or any following category name.

The main bookmark window 210 contains a notation 236 that the displayed category (in this case, the top-level category) is private. When that is the case, only the user 110 that owns (i.e.,

20   contributed) the list 212 can access those bookmarks. By selecting a "Publish" button 238 in the data entry area 218, the user 110 can make those bookmarks available to all users; otherwise, the bookmarks in the category remain private. In a preferred embodiment of the invention, a user can

14

choose to publish any category of bookmarks (or even a single bookmark or selection of bookmarks) to the community of users as a whole, or only to selected groups of users. The concept of groups will be described in further detail below.

Although the exemplary main bookmark window 210 shows the bookmarks for a single user, in a preferred embodiment of the invention, it is also possible to browse, view, and use public bookmarks, either as an entire categorized collection, or broken down via groups of users.

Several other options are available in the main bookmark window 210. A "quick add" option 240 is available; when selected, any URL or document location dragged into the window 210 will be immediately added to the list 212 without any intermediate confirmation step. The ordinary "add" function of the main bookmark window 210 will cause a confirmation window (Fig. 5, described below) to appear before a bookmark is added via the drag-and-drop method.

A search text-entry box 242 is also provided in the data entry area 218. By typing one or more keywords into the text-entry box 242 and pressing "Enter," the system will search for bookmarks containing (either in the title or in the URL) the requested keywords. By selecting a "Search Public Bookmarks" option 244, the search can be extended to all users' public bookmarks (see Fig. 4, described below).

The main bookmark window 210 also provides a command selection drop-down listbox 246; the contents of the command selection listbox 246 are shown in Fig. 3. By selecting various options in the listbox 246, the user can add bookmarks, separators, and categories; import and export bookmarks between the bookmark system of the invention and, for example, Netscape's internal bookmark system (described in the background section above), edit the user's preferences (such as password, e-mail address, etc.); log out of the system; edit a document; show a document (ordinarily

15

performed by simply clicking or dragging a bookmark, as described above); or show a directory of users (used to access a particular user's public bookmarks).

The functions available in the command selection drop-down listbox 246, as illustrated in the window 310 of Fig. 3, involve simple data manipulation and will not be described in further detail

5   (except in the context of the general operation of the system in Figs. 9-12). These functions would be easily implemented by a person of ordinary skill in the art of Web-based application programming.

As stated above, Fig. 4 is a search results window 410 illustrating a sample set of search results; the keyword entered into the text entry box 242 in the illustrated embodiment was "internet."

10   Accordingly, the search results window 410 includes a list 412 of bookmarks containing, either in the title or URL, the word "internet." The exemplary search was performed with the "Search Public Bookmarks" option 244 selected, so the list 412 can include bookmarks from all users. The category in which each bookmark in the list 412 was found is presented next to the corresponding bookmark. For example, an "Internet Info" bookmark 414 was found in some user's "Internet" category; for

15   privacy reasons, the user's identity is concealed. As with the main bookmark window 210, the user can click or drag bookmarks from the search results window 410 to access the documents referenced by the bookmarks in the list 412.

Fig. 5 illustrates an "Add New Bookmark" window 510 used by the user interface of the presently implemented embodiment of the invention. As shown, in its empty form, this window 510

20   is accessed by selecting the "add bookmark" command from the command selection listbox 246 (Fig. 2). The user 110 can then manually select a category 512 in which to add the bookmark, enter the URL or other identifier 514 for the bookmark, and enter a title 516 for the bookmark. An "Add"

button 518 is provided, and when chosen, the bookmark system adds the specified bookmark to the list 212. When the "Add New Bookmark" window 510 is accessed by dragging a link onto the main bookmark window 210, the category field 512 and URL field 514 are already filled in, and can be edited by the user if desired before confirming the action by pressing the "Add" button 518.

5   Fig. 6 represents a bookmark editing window 610 presented by the user interface of the presently implemented embodiment of the invention. This window 610 is accessed by pressing an "Edit" button 248 in the main bookmark window 210 (Fig. 2). As shown (and as represented by a scrollbar 612 on the right side of the window 610, the editing window 610 has been scrolled down to reveal the bottom of the window; the top (not seen) includes the data entry area 218 (Fig. 2).

10   A list 614 of bookmarks to be edited is presented within the editing window 610; the bookmarks are preceded by a column of checkboxes 616 available to indicate which bookmarks are to be edited. By selecting a "Select None" button 618 or a "Select All" button 620, none or all of the checkboxes 616 will be selected, respectively. Editing operations, as indicated by the remaining buttons within the editing window 610, can then be performed on the selected bookmarks. For

15 example, after selecting one or more bookmarks, pressing a "Delete" button 622 will remove the selected bookmarks from the list 614 (and from the list 212 in the main bookmark window 210). A "Move to" button 624 and corresponding category selection listbox 626 are provided to move one or more selected bookmarks to a different category. Finally, a "Move Bookmarks to Top" button 628 and a "Move Bookmarks to Bottom" button 630 are provided to facilitate moving one or more

20 selected bookmarks around within the displayed category. As with the commands shown in the listbox 246, these editing commands will not be discussed in further detail, as they represent simple data manipulation operations easily implemented by a person of ordinary skill in the art.

As suggested in the summary section above, the present invention harnesses the power of online communities to allow enhanced search-and-retrieval and recommendation operations. These operations are enhanced through the use of data derived from a collection of users' shared bookmarks.

5   Millions of people use and participate in the Internet today. However, these users have many diverse interests. For example, some users might have a particular interest in financial information, while some other users might be interested in computer programming, and still other users are interested in video games. While traditional search engines are able to search the Internet as a whole, it is not usually possible to derive information from the various communities of users around the

10   Internet. Systems such as Google, CLEVER, Direct Hit, and Alexa attempt to derive some additional information from aggregate interests manifested on the Internet as a whole, but are generally unable to derive any information from the preferences of select individuals or groups.

The present invention uses communities of users as follows. In Fig. 7, the millions of users participating in the Internet are represented by eighteen representative users 710. Two groups 712

15   and 714 are highlighted. A first group 712 contains six users; for the purposes of this example, suppose that the first group 712 is interested in financial information. A second group 714 also contains six users; for exemplary purposes, suppose that the second group 714 is interested in computer programming. It should be noted that the groups need not be (and in this example are not) mutually exclusive, as one user 716 is a member of both the first group 712 and the second group

20   714. Moreover, not every user must have a defined interest. The present invention is able to use the preferences of the entire Internet 710, one or more groups 712 or 714, or even a single user 716 to provide recommendations on preferred documents and to enhance search queries and results. The

18

operations performed in doing so will be described in further detail below in conjunction with Figs. 14 and 15.

Fig. 8 illustrates schematically a similar breakdown of documents on the Internet. The millions of documents available on the Internet are represented by 24 exemplary documents 810.

5 While there are many, many documents available, only a small portion of them 812 are typically used or accessed by a community of users. And of those accessed, only a small number are preferred and bookmarked 814. The distillation of the entire Internet down to those documents that are bookmarked 814 is a powerful two-step filter that tends to pull out only the most relevant, interesting, and valuable documents. Moreover, when a community of users' bookmarks are shared,

10 other users' efforts in locating useful documents can work to the entire community's advantage through a system according to the invention.

At the system level, there are two ways in which a system of the nature of the present invention can be implemented. First, the browser 112 can be used relatively passively, to collect user input and pass it along to the database 120 for processing. This presently preferred mode of

15 operation is illustrated as a block diagram in Fig. 9. A user's system or terminal 910 hosts a Web browser 912 such as Netscape Navigator or Microsoft Internet Explorer; the browser 912 is able to interpret both HTML and JavaScript (or another HTML-embeddable scripting language) by way of an HTML interpreter 914 and a JavaScript interpreter 916. The user's system 910 also includes an input device 917 allowing the user to interact with the system 910, as well as some storage space

20 918.

The user's system 910 and browser 912 receive display information and scripts (arrow 920) from the database 922; the browser 912 simply passes along user input (arrow 924) to the database

19

922. Essentially all processing is performed at the database 922 by way of a bookmark server 926 and a data processor 928. The details of the processing will be discussed below in conjunction with Figure 11.

An alternate mode of operation, in which some of the processing is performed by the user's

5   system, is set forth as a block diagram in Fig. 10. In this case, a user's system or terminal 1010 hosts a Web browser 1012, which in turn includes an HTML interpreter 1014 and a language interpreter 1016. The language interpreter 1016 is able to receive and execute computer programs in a high-level language such as Java; most modern Web browsers include that capability. Like the system illustrated in Fig. 9, this system also includes a user input device 1017 and temporary storage 1018.

10 In the mode of Fig. 10, the browser 1012 receives bookmark data and a shared bookmarks ("SB M") computer program (arrow 1020) from the database 1022 at the beginning of a session of use; the SBM program allows the user's system 1010 to perform many of the operations disclosed herein (except for those that require large amounts of data in the database). For example, a user's bookmark collection can be maintained and modified locally at the user's system 1010 without

15 sending data to and receiving data from the database 1022. At the end of a session of use, the modified bookmark collection (and any other collected information) is transferred back (arrow 1024) to the database 1022 for storage and manipulation by the database.

As stated above, many operations performed by the shared bookmark system can be executed at the user's system 1010 without any further communication with the database 1022. However,

20 operations that tend to access a lot of data, such as database queries (arrow 1024) for search-and-retrieve operations and recommendations based on other users' bookmarks, still are preferably transferred back to the database 1022. It is simply impractical for all shared bookmarks from a

20

potentially very large community of users to be transferred to the user's system 1010 at the beginning of each session. Accordingly, these search and recommendation operations are acted upon by the database 1022 (through the operation of a bookmark server 1026 and a data processor 1028), and results are sent back to the user's system 1010 (arrow 1020). The details of the processing performed in this embodiment will be discussed below in conjunction with Figure 12.

In both of these cases, the persistent data maintained by the shared bookmark system is kept at the database 120 (Fig. 1); the user's local storage 124 is used only for temporary storage during a session. Accordingly, if the user 110 wishes to access the Internet 118 from a different terminal 126, the shared bookmark system is still operational (provided the user is able to log in using a memorized screen name and password) because all necessary data is received from the database 120 at the beginning of a session, and sent back to the database 120 at the conclusion of the session. Any custom software required, other than the Web browser, is downloaded from the server 120 to the alternate terminal 126.

The approach set forth in Fig. 9 has some advantages, in that a complicated bookmark-processing program and a potentially large collection of bookmarks do not need to be downloaded from the database 120 to the user 110 at the beginning of each session. This reduces any lag or delay between the user initiating the service and the availability of the bookmarks, which can be an inconvenience.

The approach set forth in Fig. 10 has a different set of advantages. To the extent there are any communications bottlenecks between the database 120 and the user 110, the effects of these bottlenecks will be reduced when the user 110 is able to perform as much computation as possible, without the need to shuttle information back and forth between the user 110 and the database 120.

21

This approach will also be advantageous if the database 120 tends to be computationally overloaded. However, as stated above, the database 120 still must be accessed for certain operations.

Figure 11 illustrates the steps performed in the passive client-side embodiment of the invention (illustrated in block form in Fig. 9). The system operates by simply awaiting user input

5   (step 1110), transmitting that user input from the user's system 110 to the database 120 (step 1120), receiving a response from the database 120 (step 1130), and refreshing the display (step 1140) according to specific HTML and JavaScript instructions received from the database 120. As stated elsewhere in this document, the specific operations on the database side of the system typically represent simple data manipulation operations, which will not be described further. To the extent

10   those operations are more complicated or contain novel steps, they will be described in further detail below.

Figure 12 illustrates the steps performed in the active client-side embodiment of the invention (illustrated in block form in Fig. 10). The system operates by the user 110 initially receiving the shared bookmark program and bookmark collection from the database 120 (step 1210). The user's

15   system then awaits user input (step 1220) and processes the input (step 1230) to determine if it requires the processing resources of the database 120. If the user input comprises a query (step 1240) involving other users' bookmarks, the query is transmitted (step 1250) to the database 120 and a response is received (step 1260) from the database. The function of the database 120 at this point will be discussed in further detail below with reference to Figs. 14 and 15. If the user input is not a

20   query, the user's bookmark collection is updated (step 1270) as specified in the shared bookmark program. In both cases, the display is then refreshed (step 1280) to reflect the operation, regardless of whether it was performed locally or remotely. At the end of a session, the user's system 110

22

transmits back to the database 120 all necessary updated information, including any changes to the user's bookmark collection and any logged events (such as bookmark accesses) used by the invention to determine frequency and recency of use.

As described above, the database 120 performs certain operations in the background maintain

5   the bookmark collections of its users; those operations are set forth in Fig. 13. For reasons set forth in detail above, the database 120 monitors changes (step 1310) for each document referred to in each user's bookmark collection. This operation can be performed on a regular basis, for example hourly or daily. For the presently implemented and preferred embodiment of the invention, an icon indicates to the user whether a document has been updated within the last thirty days (see Fig. 2), a

10   daily check for updates is sufficient. To accomplish this task, the database 120 can keep either a cached copy of the document referred to (thereby enabling word-for-word comparisons between the old cached copy and an updated version), or preferably, can maintain a simple suitably robust hash value representing the document's contents, whereby any change in the document's contents will cause the hash value to change.

15   The database 120 also monitors the availability (step 1320) for each document referred to in each user's bookmark collection. This operation can also be performed on a regular basis. For the presently implemented and preferred embodiment of the invention, an icon indicates to the user whether a document is available (see Fig. 2). To derive this information, the database periodically attempts to access each document, and if an HTTP error (such as "404 Not Found") or no data is

20   returned, the document is marked unavailable for all users who have the bookmark in their collection. Unavailable documents are checked again at a later time.

23

The database 120 monitors the popularity (step 1330) for each document referred to in each user's bookmark collection by periodically determining how many users, out of the total number of system users, have a bookmark in their collection. This information is also represented by an icon (see Fig. 2). As stated above, various other metrics of popularity are trackable by the system, such as

5 frequency and recency of access. Every time a bookmark is used by a user to access a document, the time, date, and nature of that access is logged. Accordingly, the system is able to use various popularity metrics in the enhanced searching and recommendation operations discussed below.

In one embodiment of the invention, metadata is extracted from documents by the database 120 (step 1340). In the case of HTML documents, such metadata may include information on the

10 date and time a document was created, the author of the document, search keywords, and many other possible items of data that are typically concealed when a user views a document. This information can optionally be kept with bookmarks in the database 120 to facilitate further search options.

In an embodiment of the invention, the contents of documents referred to by users' bookmarks are summarized (step 1350) and stored at the database 120. Alternatively, if

15 summarization is not performed in the background, it can be specifically performed at the time of a search or upon a user request for a summary. Several summarization techniques are well known in the art. A particularly advantageous summarization method is set forth in U. S. Patent No. 5,638,543 to Pedersen et al., entitled "Method and Apparatus for Automatic Document Summarization," the disclosure of which is hereby incorporated by reference for all it teaches as though set forth in full

20 herein. In either case, document summaries optionally can be presented to users as part of a list of search results or within a list of bookmarks, enabling the users to better and more easily determine which documents from a collection are most relevant to their particular interests.

A similar technique can be used to summarize a group or collection of documents. Upon a user's request (this would not typically be performed in the background), a document summary can be generated based on a concatenation of selected documents, or even all documents referenced in a category of bookmarks.

5   The database 120 also maintains itself (step 1360) in the background by analyzing users' bookmark collections, eliminating duplicate bookmark entries as necessary, and performing additional tasks (such as "garbage collection") well known in the art of computer systems.

Finally, as discussed above, the browsing, searching, and bookmark-using habits of users of the shared bookmark system of the invention all contribute additional information to the system for

10   use in improving and refining user group profiles. For example, if a user in a computer programming group searches for, views, and ultimately bookmarks several sites relating to, say, the Objective C programming language, then the user's profile will be automatically updated to include information derived from the bookmarks, and the computer programming group's profile can also be similarly updated. This is accomplished, as will be described in further detail below, by calculating a content

15   vector representative of the bookmarks in a user's (or group's) collection.

The concept of groups, as generally discussed above in connection with Fig. 7, is central to the present invention. While the embodiment of the invention disclosed and described above, particularly in the user interface windows of Figs. 2-6, does not employ the concept of groups for many operations, it should be recognized that organizing the shared bookmark system's users into

20   groups is a powerful way to capture preferences and relevant documents.

For example, in the user directory introduced above in connection with Fig. 3, there can also be group membership information. In an embodiment of the invention, a user can manually choose

to become a member of a group having similar interests (as evidenced by manually inspecting the users' public bookmarks), or in an alternative embodiment, can be assigned automatically to groups based on clustering on the user's public bookmarks (or, preferably, the contents of the documents pointed to by the bookmarks). For a detailed description of methods of document clustering, see

5   U.S. Patent No. 5,442,778 to Pedersen et al., entitled "Scatter-Gather: A Cluster-Based Method and Apparatus for Browsing Large Document Collections," the disclosure of which is hereby incorporated by reference for all it teaches as though set forth in full herein; see also U.S. Patent No. 5,659,766 to Saund et al., entitled "Method and Apparatus for Inferring the Topical Content of a Document Based Upon its Lexical Content Without Supervision," the disclosure of which is also

10   hereby incorporated by reference. One method of automatically grouping users involves identifying the centroid related to each user's public bookmark vectors (either collectively or on a category-by-category basis) in document space, and collecting the mutually-nearest sets of users into topic-related groups. Alternatively, a less-automatic method includes computing a centroid, as above, but using that information simply to recommend related groups to the user.

15   Another possibility, related to the foregoing automatic grouping scheme, is to create virtual groups based on the topic categories in each user's public bookmarks. For example, if a user has a public bookmark category entitled "Java," that user can be treated as belonging to a group having an interest in the Java programming language.

As stated above, each user and group (however created) can have a profile associated

20   therewith. In the preferred embodiment of the invention, a user's profile consists simply of a normalized content vector representing the aggregate contents of all of the user's public bookmarks. For a description of how this vector is calculated, see U.S. Patent No. 5,442,778 to Pedersen et al.,

26

described above. Similarly, a profile for a group includes a normalized content vector representing the aggregate contents of all public bookmarks belonging to the users within the group. The user and group profiles are used in the search and recommendation aspects of the invention, which will be described in further detail below.

5   Referring now to Fig. 14, a recommendation service according to the invention is set forth as a flow chart. Preferably, this service is initiated by a user by selecting a command in the main bookmark window 210 and entering text representative of the desired subject (step 1410). The user then identifies the context (step 1420) within which the recommendation should be generated. This step is preferably performed by manually selecting a user or group profile, which as described above, has a content vector associated therewith. If no existing single user or group is satisfactory, a special-purpose group can be assembled by the user by manually selecting users and having those users' profiles merged into a special-purpose content vector. The user then selects a level of "relevance feedback" (step 1430). Relevance feedback allows the user to select whether the desired documents are those similar to the selected context or dissimilar to the selected context. A known example of positive relevance feedback is the "more like this" option provided by the Excite search engine.

The subject, context, and relevance feedback are then processed by the database 120 to generate recommendations (step 1440). This step uses the popularity (proportion of users having a bookmark) metric 1450, frequency of use metric 1460, and recency of use metric 1470 described 20   above. The recommendation generation step searches the public bookmarks belonging to the user or group selected as the context (or alternatively, all public bookmarks) for the keywords identified as the subject. These keywords can be found in the title or URL of the public bookmarks, or

27

alternatively, in the content belonging to the public bookmarks. The matches are then analyzed with respect to the content vector and the popularity, frequency, and recency metrics. Alternatively, the keywords are combined with the content vector into an enhanced keyword vector, which is then compared to the entire search corpus in a single search step and then ranked according to popularity,

5   frequency, and recency.

In an alternative embodiment of the invention, the recommendation service is also able to use link structure (as described above in connection with the Google and CLEVER systems discussed in the background section above, for example) in generating recommendations. By way of example, the number of inlinks or outlinks (weighted or unweighted), as well as any other link-specific metrics

10 readily apparent to those skilled in the art, may be weighted and incorporated into the step of generating recommendations (step 1440) via the content vector, popularity, frequency, and recency.

Alternatively, a spreading activation methodology may be used to re-rank recommendations (or search results, as will be described below), or may be used as a weighted factor in the recommendation operation set forth above. See U.S. Patent No. 5,835,905 to Pirolli et al., entitled

15 "System for Predicting Documents Relevant to Focus Documents by Spreading Activation Through Network Representations of a Linked Collection of Documents," the disclosure of which is hereby incorporated by reference for all it teaches as though set forth in full herein. Spreading activation techniques are based on representations of Web pages as nodes in graph networks representing usage, content, and hypertext relations among Web pages. Conceptually, activation is pumped into one or

20 more of the graph networks at nodes representing some starting set of Web pages (i.e. focal points, which in the context of the present invention may be a set of highly ranked recommendations or search results) and it flows through the arcs of the graph structure, with the amount of flow

28

modulated by the arc strengths (which might also be thought of as arc flow capacities). The asymptotic pattern of activation over nodes will define the degree of predicted relevance of Web pages to the starting set of Web pages. By selecting the topmost active nodes or those above some set criterion value, Web pages may be aggregated and/or ranked based on their predicted relevance.

5   If the user selected positive relevance feedback (i.e., "documents like these"), the closest context matches in the previously-keyword-matched public bookmark collection are returned as the highest-ranking. If the user selected negative relevance feedback (i.e., "documents unlike these"), the closest context matches in the collection are given the lowest rankings. The recommendation list is then returned to the user for viewing (step 1480).

10   It should be noted that recommendations need not be provided only in the context of an explicit request for recommendations in through the main bookmark window 210; recommendations can also be provided while a user is browsing the Web. If a Web page the user is viewing contains a link that is also a popular link (by any of the metrics defined above) in the shared bookmark collection (as a whole, or within one or more groups), then the user can be alerted to that via the presentation of a message in the browser window. Alternatively, if there is a historical pattern of documents chosen from the shared bookmark collection (as defined by the frequency and recency metrics), then that pattern can be highlighted for the user.

One embodiment of the recommendation service is adapted to provide a user with a "substitute" bookmark when a preferred document is unavailable, as indicated by the unavailability icon 228 (Fig. 2). This capability is implemented through a variation on the recommendation service set forth in Fig. 14. Specifically, instead of entering text representative of the desired subject (step 1410), the user identifies an unavailable bookmark. Information stored by the database 120

29

pertaining to the unavailable bookmark is then used to generate a recommendation; that information may (in various embodiments of the invention) include a content vector, keywords, or a summary of the expected (but missing) content.

A search and retrieval system according to the invention is set forth in Fig. 15 as a flow chart.

5   As with the recommendation service of Fig. 14, the user begins by formulating a keyword query (step 1510), identifying a context (step 1520), and identifying relevance feedback (step 1530). The query, context, and feedback are provided to the database 120, where the query is augmented (step 1540). In a preferred embodiment of the invention, query augmentation is performed by selecting the most important (i.e., highest magnitude) words from the context's content vector and adding them to

10   the query. It should be noted that various other methods of augmenting the query with the user's preferred document collection are possible, as suggested above, ranging from the decomposition of the entire collection into a single vector, the selective decomposition of portions of the collection based upon similarity or dissimilarity to the initial query or based upon groups, to completely augmenting the query with document collection.

15   If positive relevance feedback (i.e., "documents like these") has been selected, the query is augmented by adding the additional context words as words that *should* be found in the results; if negative feedback has been selected, the query is augmented by adding the additional context words as words that *should not* be found in the results. It should be observed that the user's initial query is being augmented by introducing additional search words for a simple reason: traditional search

20   engines generally operate only on search terms, and would not recognize a content vector, context, or information in any other form. While various search engines use different syntaxes for specifying

30

positive and negative relevance feedback, in most cases, some degree of augmentation according to the invention is possible.

The query is then performed. In a preferred embodiment, the query is performed on the database 120 of public bookmarks (step 1550) at the same time it is performed on an external search

5   engine (step 1560). If the community of users is sufficiently large, results obtained from the public bookmarks may be superior in quality to those received by querying the Internet; this approach is similar to the recommendation system described above (Fig. 14). The search results obtained from the database 120 can be presented to the user separately from or as a part of the search results obtained from the Internet.

10   The results are then ranked (step 1570) according to the user's selected context, as with the recommendations described above (see step 1440). Also, the user may be given the option to employ popularity, frequency, and recency in the ranking operation. And in an alternative embodiment, as described above, link structure may also be used.

The relative weights of the context match, link structure, popularity, frequency, recency, and

15   any other metrics in the ranking operation are matters of preference, and in a preferred embodiment of the invention, can be adjusted by the user. In an alternative embodiment of the invention, the weights are dynamic and adjustable by the system based on user habits. For example, if a user consistently chooses relatively low-ranked selections in search results lists, that may be seen as an indication that the current metric weights are incorrect, leading to a de-emphasis of the highly

20   weighted factors. Such learned weightings can be maintained as global system defaults or, preferably, applied on a user-by-user basis.

31

Alternatively, as described in conjunction with the recommendation service above, a spreading activation methodology may be used to re-rank search results, or as a factor in the ranking operation set forth above.

The ranked results are finally returned (step 1580) to be viewed by the user. As stated above, the results retrieved from the public bookmark collection can be presented separately from the Internet results, or they can be incorporated into the same list (most likely with higher scores, because of influence from the popularity metric). In a preferred embodiment, the results retrieved from the public bookmark collection are highlighted to identify their origin.

It should be noted that a search service according to the invention need not provide both query augmentation and results ranking. Advantageous results can be obtained by using either technique alone. For example, a query can be augmented by a system according to the invention, after which Internet search engine results are viewed without further modification. Or alternatively, a user-formulated query can be passed directly to a search engine, with the results being passed through the context and popularity-based ranking scheme of the invention.

The search and retrieval system discussed in connection with Figs. 14 and 15 has particular utility when it is used in conjunction with third-party search and retrieval systems, such as the Google information search and retrieval engine, commonly used to examine information repositories such as the World Wide Web. As was described previously, contemporary search and retrieval systems are generally limited in their ability to effect a true understanding of the type of information being sought by a user, particularly when the information search request is presented as a simple key word query. Generally, a user will provide a key word, or a set of key words, to the search engine which, in turn, attempts to match the search key words with some form of content or content

designations, representing the various documents available through the World Wide Web. Search engines make use of various weighting techniques, all of which are well understood by those having skill in the art, in order to attempt to extract relevant documents from the information pool in accordance with a key word query. However, by being unable to contextualize, search engines are

5   very limited in their ability to handle result diversity, where documents' content might contain all the relevant key words, but represent a topic totally out of context.

For example, performing a key word search for "java" might naturally result in a search engine's returning a vast number of documents relating to the JAVA programming language, with only a relatively few documents relating to java the coffee, or Java the island. If the relevant user

10   were more interested in obtaining information on java the coffee, particularly grown on Java the island, they would have to wade through a collection of perhaps thousands of documents relating to JAVA the programming language before reaching their goal.

The system and method of the present invention provides unique query augmentation and contextualization features that can be incorporated within, or provided as an adjunct to, the various

15   conventional search and retrieval systems that are presently employed. Combining the inventive system with a conventional search and retrieval engine is particularly advantageous when it is realized that the World Wide Web comprises an information repository, which might be considered as organized in accordance with a rational structure, the use of which might enable the researcher to more efficiently extract relevant information. In the case of the World Wide Web, a suitable

20   hierarchical structure might be that defined by ODP, which is a well known and well understood classification model for Web content. If the Web were considered as a branching nodal structure, ODP appropriately labels the nodes, from top level domains to the bottom, with content descriptors,

33

in order that the hierarchical structure can be traversed, from the more general to the more specific, by a perceptive user. Implicit in this conceptualization of the Web as a hierarchical structure, is the realization that a user need not begin at a top level node in order to traverse the hierarchy and reach a specific sub-level that contains the information they are seeking. A user familiar with the ODP

5   structure is able to conduct a search directly within a particular sub-category, provided they are aware of the existence of that sub-category. It should be appreciated, therefore, that the system and method of the present invention allows for a significant degree of query augmentation and contextualization to be added to a conventional key word query search, such that a user is not compelled to remember or even be aware of the identity of a particular sub-category in order to avail

10   themselves of the hierarchical structure comprising the Web.

In one particular embodiment, the system of the present invention is interposed between a user's terminal or interface device and a search engine, where it functions as both a query augmentation engine and as a contextualization filter. As shown in the exemplary embodiment of Fig. 16, the novel system is identified as a relevance engine and is identified generally at 1601. The

15   relevance engine 1601 is functionally interposed between a user 1602 and is configured to receive key word query information from the user and provide a filtered response back to that user. The relevance engine 1601 is further coupled to a search engine 1604, to which it provides an augmented query and from which it receives conventional search results. Search engines are conventional in nature and need not be described further herein, except to mention that the engine accesses the Web

20   1606 and extracts what it feels to be relevant information therefrom in accordance with matching parameters defined by a received query. In the case of the Google engine, the search engine 1604 is able to provide some degree of recommendation ordering to the search results, but that

34

recommendation ordering is based solely on relevance metrics defined by the Web community as a whole. In a manner that will be described in greater detail below, the relevance engine 1601 provides additional relevance filtering to the results returned by the search engine 1604 and offers the filtered search results (either documents, URLs, or some other form of object) to the user 1602.

5   Similarly, and in accordance with an additional embodiment of the invention illustrated in Fig. 17, the relevance engine 1701 is interposed between a user 1702 and a search engine 1704 and is configured to receive key word query data from the user and pass, not only an augmented query, but also a contextualized query, to the search engine 1704. In this particular embodiment, context information is provided as a content vector which might, in turn, be made up of individual profile information, group profile information, or a relevance vector derived from an individualized hierarchical relevance model, as will be described in greater detail below. Search engine 1704 is able to internalized the augmented query and contextualization information and conduct a search of an information repository 1706 such as the Web with greater efficiency and extract a collection of objects, such as documents, that contain a greater degree of relevance to the user 1702.

15   In either of the embodiments mentioned above, the relevance engine 1601 or 1701 might be allocated between a user's terminal and database server in the manners described in connection with Figs. 9 and 10, above. Indeed, it is not particularly material how the relevance engine is allocated between a user's terminal and a database server, so long as the component portions of the relevance engine are so disposed so as to receive relevant input from a user, process that input in accordance

20   with the invention, and pass the results of such processing to a search engine.

The system might be implemented, for use in connection with a conventional search engine, by having the search engine's website contain an offering of the novel system, which might be

accessed by a user by depressing a button bar, activating a pop-up window, traversing a link, or the like. As a user activates a link to their favorite search engine, and as the search engine's main screen appears, the user is given the opportunity to subscribe to the novel system and, after having done so, the system is now available for use in conjunction with the user's chosen search engine. Relevant

5   portions of the system are downloaded to the user's computer in a manner described above, and gives the user the same functionality and capability as was described above in connection with Figs. 2-6. The user may organize selected bookmarks (typically found in a user's Favorites folder) into a user-defined categorical structure, join a group and import either all or some portion of that group's hierarchical categorical structure, or both.

10   In addition to creating a personalized categorical relevant structure, the user is also requested to complete a personalized data form which will be subsequently used as the basis for a personal profile script unique to that user. The profile form contains data fields into which certain customary demographic information might be entered by the user, such as age, gender, geographic place of residence, as well as certain particularized interest fields that the user may optionally designate.

15   These personalized interest fields correspond to the top level categorical node designators of the categorical hierarchy of the directory system used by the particular search engine in question. In the example of the Google search engine, the particular interest fields might be those 12 to 15 top-level categories identified by ODP, i.e., Sports, Entertainment, News, Business, and the like. Thus, each individual user's personalized profile information is set forth in a manner which is related to the

20   categorical structure of hierarchical model of the information repository (the Web) used by the search engine.

Once the system of the invention has been accessed, and the user has completed the profile file, the system may be used in conjunction with a search engine in order to provide more directly relevant search results to that particular user. As was described above, the browsing, searching and bookmark-using habits of users of the system all contribute additional information to the system for

5 its use in improving and refining each user's profile information. Additionally, as groups are identified to the system, the system is able to use group browsing, searching and bookmark-using habits in order to define and refine that group's profile. In this particular instance, profile development for both individual users and groups, proceeds in a manner as described above, with the user and/or group file being automatically updated to reflect the changing interests and preferences of

10 profile membership.

In the simplest case, a user or group profile comprises a normalized content vector representing the aggregate contents of all of the bookmarks, whether public or private, pertaining to that user or group. Additionally, the profile represents not only the content of a user or group's content collection but also the structure or organization of that content in accordance with the user or

15 group categorical hierarchy scheme. Content contained within a particular categorical designation is processed by the system, as described above, and associated with that categorical hierarchical designation as a further component to the profile's content vector. In this manner, the system builds up a relevance profile for each individual and/or group and maps that profile in accordance with a determined relevance model to collection content.

20 As will be appreciated by those having skill in the art, individual relevance profiles, relating to information content, can be mapped to a global hierarchical structure, such as the ODP categorical model into which Web documents are organized. All that is required is a knowledge of hierarchical

categorical structure of the Web organizational model and a training set of documents which are each assigned to particular categorical nodes within the structure. The categorical designators for the nodes and the content of the documents associated to each categorical node provides the correspondence metrics between the global hierarchical model (the Web) and a user or group's

5   relevance model represented by an individual or group profile.

In one particular embodiment, the individual or group profile (also termed a relevance model) defines a simple interest model for the individual or group to which that profile pertains. A user's choice of particular bookmarks and the categories to which those bookmarks might be assigned will be understood to represent that user's hierarchical content interest model and that the user's interest

10   model represents a subset or overlay to the global hierarchical model defining the Web. Relevance data is mined from each user's profile information and particularly from the structure, organization and content of that user's bookmark collection.

The system according to the invention functions, in conjunction with a search engine, to provide augmented query information to the search engine and further to function as a relevance

15   filter to information returned by the search engine from an information repository. In operation, the user initiates a key word query search by typing the search key words into a data entry field and forwarding the query to the inventive system. The inventive system augments the query with information derived from that user's profile information and additionally, with information derived from where that user might be positioned within the user's information space when initiating the

20   query. For example, if the user were a member of a computer programming group and the user's activity log placed the user within the "computer programming" category when the search was initiated, the system would augment a query for the term "java" with additional key words associated

to content derived from the "computer programming" category. Even though the user might have an interest in Java the island, the user's location within the computer programming category would suggest an immediate interest in JAVA the programming language with respect to the user's key word query search. In this particular instance, the system is able to recognize a "temporal" interest

5 expressed by the user in JAVA the programming language, due to the user's spatial location within their own categorical hierarchy when the search was initiated. The user's location within their categorical hierarchy defines a content vector which is added to key word query terms and passed to the search engine as an augmented key word query. The search engine, in turn, uses the augmented key word query to examine the collection of documents compiled in the information repository,

10 computer match thereto, and return a more relevant set of documents to the user than what might be expected from the initial, simple key word query. In this particular instance, it will be appreciated that the key word query augmentation performed by the system is a type of contextualization that implicitly recognizes a user's immediate interest depending on a user's spatial positioning within their categorical hierarchy. Positioning might be with respect to the user's individual categorical

15 hierarchy, as represented by the structure and organization of their own bookmark collection, or alternatively, by the user positioning within a group hierarchy when initiating the search. The system recognizes that a user's initiating space is often highly relevant to the next form of information that they might wish to receive. This is analogous to the "hub" methodology used by various websites, wherein a particular site might include links to various other, related sites which comprise content

20 that is relevant in one degree or another to the content provided by the "hub" site.

It should be understood that the system according to the invention is not limited to providing augmentation of key word queries depending upon the location of a user within the user's categorical

39

hierarchy when initiating the query. Indeed, a user might wish to initiate a key word query even if the user is not accessing the system by positioning themselves within either their categorical hierarchy or that of a group. In this particular case, the user is deemed to be positioned within their top-level domain and, accordingly, the user's profile information forms the basis for the query

5   augmentation. In this particular instance, the user's entire bookmark collection might be decomposed into a single content vector. The selective decomposition of portions of the collection based upon similarity or dissimilarity to the initial query augments the query with suitably derived content vectors from the document collection. Once this augmented query is passed to the search engine, the search engine searches the information repository and returns results that match or

10   correspond to the augmented key word query terms.

Once documents have been returned, the system according to the invention functions as a recommendation engine in order to filter the returned search results and provide the filtered results to a user in keeping with the degree of relevance of each document to the user's initial query. Certain forms of recommendation engines are well understood by those having skill in the art, but all of these

15   conventionally recognized recommendation engines rank or order documents on the basis of a relevance metric established by the entire Web-using community as a whole. None of the conventionally understood recommendation engines are able to deal with result diversity of the kind that frequently obtains when queries, even augmented queries, might pertain to totally different subject matter. In accordance with the practice and principles of the invention, the relevance engine

20   is able to order or rank results return from a conventional key word search in accordance with a model based on a "standard directory model of the universe" such as represented by ODP. In accordance with this particular model, a user's search results are evaluated for their allocated

40

positions within standard directory model and are allocated a content vector accordingly. This content vector identifies the category in which such a document was found, along with sufficient content information relating to that document so as to position it, with fine-grained detail, within the global model comprising the corpus of information. Such information may be mined from the

5   content of the document itself, from a document's metafile information or some other content designator or descriptor that might be allocated or assigned to that document by the entity in charge of maintaining the global model. Any document's relevance to a particular user's search is discerned by comparing its hierarchical or categorical content information, in conjunction with its content, to not only the query or augmented query which initiated the search, but also to an interest

10   categorization or classification model developed by the system with respect to each user. The interest categorization model (or interest classifier) is a personalized hierarchical interest model constructed in accordance with a system and method described in a co-pending patent application entitled HIERARCHICAL MODEL FOR DYNAMIC PERSONALIZATION OF OBJECT SPACES AND PRIVACY PROTECTION, filed on instant day herewith and commonly owned by the

15   assignee of the present invention, the entire contents of which are expressly incorporated herein by reference. In particular, a user's choice of categorical organization for their personal bookmark collection, as well as the titles and/or descriptor fields used to denote the categorical nodes of the collection provide a highly reliable inference as to which fields of information that particular user is likely to find relevant. All other user's individual categorical information, as well as categorical

20   information pertaining to groups which that user has elected to join, is used by the system to develop and refine each user's personalized relevance profile. Since each of the nodal points of either a user's or a group's categorical structure are defined by the content contained within a node, it is a

relatively simple matter to map a user's node name to a particular node of the standard directory which contains corresponding content. It should also be noted that the words or phrases chosen to identify the nodes will give an indication of the context in which the user views the content contained within those nodes.

5   Given this personalized relevance model, search results returned by a search engine may be filtered by comparing the content of the documents to a content vector derived from a user's profile. In addition, the category from which those documents were extracted is compared to the categorical structure of the user's collection and, documents extracted from corresponding categories might be given additional weight as relevant documents. Documents extracted are then ordered or ranked in

10   accordance with a degree of correspondence or matching between the document's category and content and the content and categorical (contextualization) metrics derived from a user's interaction with the system. As standard directory model categories are found to contain content corresponding to content contained within a user's categorical model, the standard directory model category is mapped to the user's equivalent category. This mapping is then stored in the user's profile for future

15   reference.

The foregoing description of query augmentation and contextualization is an example of what might be termed topical or categorical contextualization. It is an additional feature of the present invention that it is able to inform the relevance characteristics of a search by time or temporal contextualization as well. As was described above, and is described in greater detail in co-pending

20   patent application entitled HIERARCHICAL MODEL FOR DYNAMIC PERSONALIZATION OF OBJECT SPACES AND PRIVACY PROTECTION, the system is able to monitor a user's activity when the user is engaged with the system. For example, as a user surfs the Web, the system monitors

the user's click stream and stores the historical series of URLs which defines a user's path through the Web during a session. Irrespective of whether or not the user decides to save any of these URLs as a bookmark, the mere activity of the user in accessing these URLs reliably implies some degree of interest by the user in the content represented thereby. Relatively short-time user activity therefore,

5 is a reasonable indicator of "interest immediacy," and quite likely points to a developing interest in a new information field that is unrepresented by a personal profile or a personal hierarchical relevance model, as currently implemented. This particular feature will be recognized as advantageous since the meaning of identical words or terms can contextually change with time. For example, a computer programmer who would normally be recognized as placing relevance upon "JAVA" the

10 programming language, might be interested in planning a vacation to "Java" the island. That user might have recently traversed a number of travel-related sites, such as airlines, travel agencies, and the like, looking for information on booking such a vacation. Additionally, that user might have traversed a number of sites relating to Java the island. Combining this user's recent interest in travel along with their recent interest in Java the island, a subsequent key word search for "java" would

15 imply the necessity of some measure of result differentiation. Given the recentcy of "travel" and "Java" in the user's surf and search focus, the system is able to place a recentcy weighting on the topic "travel" and the document content "Java" and "island." Accordingly, although one would expect a substantial number of high ranking documents returned by the search to be concerned with JAVA the programming language due the user's profile, a substantial number of documents will also

20 be returned pertaining to travel to Java the island, due to the user's recentcy of interest in those topics.

43

Thus, both topical and temporal relevance information can be data mined from the structure, organization and usage of the system and can be formed into a personalized set of relevance vectors that are unique to each user. Topical and categorical information contained within each user's relevance vectors are mapped to corresponding categorical nodes of a standard directory model, such

5 that there is a relevance correspondence between content in an information repository and a collection of relevant content maintained by a user.

Just as the relevance engine described in connection with Figs. 16 and 17 might have its components allocated between a user's terminal and a database server, so might the functionality of the relevance engine be allocated between the relevance engine and a search engine. For example,

10 and in accordance with the exemplary embodiment of Fig. 18, the relevance engine might contain all of the functionality required to both augment a key word query and to provide context based relevance filtering of results returned by the search engine. In the exemplary embodiment of Fig. 18, the user provides an initial key word query to the system which processes the user input and augments the key word query in accordance with the user's relevance model, a simple interest model

15 or the hierarchical content model, as described above. In accordance with this methodology, the system develops an additional set of descriptive key words other used to augment the initial key word query search. The augmented query is then passed to the search engine which searches and evaluates the contents of an information repository, such as the Web in conventional fashion. A response is received from the search engine, which response might include a set of documents ranked in

20 accordance with the search engine's own recommendation scheme. The relevance engine receives the search results and re-ranks or re-orders the results in accordance with contextualization information developed from the user's personal relevance profile, the user's immediacy metrics, the

44

user's position and/or activity while engaging with the system, and the like. These re-ordered, contextualized results are then presented to the user over the user's browser screen.

In an alternative embodiment, as illustrated by the exemplary flow diagram of Fig. 19, portions of the system's functionality can be incorporated into a search engine such that a search

5 engine is able to more efficiently derive relative content from an information repository such as the Web. In this particular embodiment, the user initiates a key word query search, in conventional fashion, through the system application appearing on the user's browser. As in the embodiment described above, the system augments the user's key word query with additional key words extracted from the structure, organization and content of the user's content pointer collection hosted by the

10 system. In addition, the system adds contextualization information to the key word query terms in the form of directed content vectors and passes the entire contextualized query to the search engine for execution. Contextualization information is derived from not only the user's structure, organization and content of the user's content pointer collection hosted on the system, but also from other aspects of the user's personalized relevance model, such as the user's immediacy interest,

15 certain document recentcy metrics, certain document frequency metrics, and the like. It should also be noted that although the contextualization content vector has been described as being added to the key word query, the contextualized content vector may just as easily include the key word query information, whether augmented or otherwise, so as to define a single contextualized content vector which is passed to the search engine for execution. In situations where the user has engaged the

20 system for a certain period of time, the system also translates any user-defined categorical descriptors into their corresponding standard directory model categorical states, as they have been so mapped by the system through usage. The content vector is then utilized directly by the search engine to filter

45

and extract only relevant documents from the information repository, i.e., only those documents meeting the full criteria of content as well as context match. In other words, the contextualization portion of the search vector tells the search engine what portion of the standard directory to access, while the augmented query portion of the search vector tells the search engine just which documents

5 in that portion of the standard directory are relevant and which pointers to that content should be extracted for presentation to the user. The search engine then returns an explicitly relevant set of content pointers to the user through the relevance engine, which may or may not perform additional filtering, such as relevance feedback filtering, at the option of the user.

Given that the system recognizes both individual users and user communities, termed groups herein, it should be understood that a particular user might have access to more than one personalized

10 relevance model for interaction with a particular information repository, such as the Web. The user may have their own relevance model which is invoked if the user logs onto the system as an individual or engages with the relevance model from an electronic space which the system recognizes as "individual." If the user were to enter a "group" space, by perhaps clicking on a group

15 or community bookmark icon, or by taking any one of a number of defined user interactions which put the user into a "group" space, this space is identified to the inventive system when a search or recommendation procedure is requested and the corresponding "group" relevance model is invoked.

Having now described the uses and utilities of the novel personalization and relevance determination system according to the invention, it would be beneficial to discuss in greater detail the

20 structure, organization and content of a user's content pointer collection might be mapped to corresponding portions of a "standard directory model of the universe." In this regard, it is necessary to begin with a structure map of the global hierarchy with which a user's hierarchical structure is to

46

be associated. As was described above, such a structure map can be an importation of a conventionally developed hierarchical structure, such as the ODP categorization set for the World Wide Web. In order to assign relevance indicia to content extracted from a particular information repository, it is advantageous for the system to have access to a training set of objects which have

5 already been classified into the structure. This last is important since from the universe of objects available through the World Wide Web, for example, only a very small fraction of the available objects (i.e., documents, content pointers, and the like) have been categorized or classified, even within the ODP structure, which is acknowledged to be by far the largest. Accordingly, a great deal of recovered content will be unclassified and will be need to be categorized into the hierarchy before

10 any personal relevance contextualization can be applied to such objects.

From the object training set, i.e., from the over one million documents classified into the ODP hierarchical structure, the system develops a database of content key words, category vectors, and other relevant information that might be used to assign or allocate a particular object into a particular structure category. Performing this content analysis on a content training set which has

15 already been categorized, the system is able to match content to category and develop its own "quasi-relevance" model for the categories which comprise the standard directory model. In effect, for each ODP category, the system retrieves classified (categorized) pages within that category, strips all textual and contextual information those pages and populates a database item associated to that category. The process is repeated until all of the ODP categories have an associated database item

20 identified to that category, such that a set of indicia is defined and linked to each category, so that each category's content is well characterized.

As unclassified (uncategorized) objects or documents are retrieved from the information repository, it is understood that they will at least contain a match with a query key word, if not a match with some contextual indicia that they have been provided by the query search initiator. For each uncategorized page, the textual information and/or contextual information contained in the

5   object is stripped from the object and compared to the contents of the various database items which characterize the global hierarchy. Once an uncategorized object's content is processed and a suitable category match is found, the object is assigned to that category and the categorical assignment is maintained for that object for all future purposes.

Thus, once an uncategorized content has been classified by the system, the object may not be

10   processed for relevance. All content, no matter how derived, can be analyzed for relevance, so long as the content (or object) is extracted from an information repository organized in accordance with a defined structure, and there exists a training set of content pages or objects which have already been classified or categorized within that structure. Any and all content (or objects) can be analyzed with regard to the results of the training sequence and can be efficiently and rationally allocated to a

15   categorical position without further invention by any other authority.

One further embodiment of the invention incorporates automatic document categorization. If this feature is implemented, a user is able to create a hierarchy of objects (whether bookmarks, document pointers, or the like) without having to manually create the categories. Several topical categorization methods are known in the art; for example "k nearest neighbors," "support vector

20   machines," and "winnow," among other methods, can be used. For several detailed examples of how document categories may be formed, see U.S. Patent No. 08/842,926 to Pirolli et al., entitled "System for Categorizing Documents in a Linked Collection of Documents"; U.S. Patent No.

5,526,443 to Nakayama. entitled "Method and apparatus for highlighting and categorizing documents using coded word tokens"; and U.S. Patent No. 5,687,364 to Saund et al., entitled "Method for Learning to Infer the Topical Content of Documents Based Upon Their Lexical Content." With the automatic categorization feature of the invention enabled, to add a new bookmark, a user need only

5   drag a link from the Web browser onto the main bookmark window 210; the automatic categorization method will ensure that the new bookmark is properly categorized within the user's hierarchy (creating a new category, if necessary). Optionally, the user can specify whether the new bookmark is to be private, shared with a group, or public; the categorization method will then act accordingly.

In the embodiment of the invention, described, above, the automatic categorization scheme

10   uses, as a topical reference, a global hierarchy of categories. An object or collection of objects is compared (by the scheme described above) against the possible topics or categories in the global hierarchy and the user is presented with a resulting categorical name that "fits" properly within the global hierarchy. This approach has the particular advantage of ensuring that searches initiated by members of a group, from within a "community" space, return consistent categorically

15   contextualized results. A document or collection of documents is compared (by the scheme described above) against the possible topics in the global hierarchy, and the user is presented with a resulting category name that "fits" properly within the global hierarchy. This approach has the advantage of ensuring that searches across multiple users' public bookmarks return consistent category names.

20   Deriving the global hierarchy can be accomplished as follows. First, represent each category as the centroid of the titles of the documents (or, preferably, the content of the documents). The centroids of the categories are then clustered into a preferred number of top-level clusters (e.g., ten

49

clusters). Then recursively cluster each of the clusters until the "leaves" of the hierarchy are individual clusters.

While certain exemplary embodiments of the invention have been described in detail above, it should be recognized that other forms, alternatives, modifications, versions and variations of the

5   invention are equally operative and would be apparent to those skilled in the art. The disclosure is not intended to limit the invention to any particular embodiment, and is intended to embrace all such forms, alternatives, modifications, versions and variations.

Accordingly, while this specification, for the sake of clarity and disclosure, at times uses specific terminology and constructs to refer to certain aspects of the invention and its operating

10   environment, it will be recognized that the invention set forth herein is applicable in other areas, as well. For example, this specification frequently refers to the Internet, Web sites, Web pages, and documents; it should be observed that the invention is equally applicable to other types of documents, databases, and document collections. Moreover, references to bookmarks, favorites, and preferences are not intended to be limited to any particular implementation (or set of

15   implementations) for retaining information on users' browsing habits, but instead should be construed to apply to all means and methods for specifying and retaining such information.

Similarly, HTML is described as the most common format or language for describing documents on the Web; it should be noted that other document formats (such as XML, SGML, plain ASCII text, plain Unicode text, and other standard and proprietary formats) are also in use on the

20   Internet and in various other document-based applications; this invention will function equally well in the context of networks utilizing other formats or even multiple formats. For the purposes of certain aspects of the invention (such as summarization and recommendation), the only limitation is

50

that the format be decomposable into a language (which can even be accomplished, in image-based formats, through character recognition). The term "document" is intended to refer to any machine- or human-readable data file (or collection of related files) from which information can be retrieved.

URLs are typically used to access information on the Internet, and frequently on other

5   networks, as well. However, it should be recognized that other means of specifying the location, identity, and nature of a requested document are also possible; such alternative schemes would be apparent to a practitioner of ordinary skill in the art, and the invention is deemed to cover these variations.

When the present disclosure refers to Web browsers, it should be recognized that other

10   information access applications are also relevant, including but not limited to information sharing and access tools such as Lotus Notes, database systems, and other data sharing and retrieval applications.